10-047-009 (REV. 6/63)

DIVISION	SNAP-8
TM	4921:66-375
DATE	8 November 1966
W.O.	1040-02-2320

TECHNICAL MEMORANDUM

AUTHOR(S): A. B. Burgess (Aerojet-General Nucleonics)

TITLE: SCAN, A Computer Code for SNAP-8 System Analysis with Influence Coefficient Calculation Option

ABSTRACT

A digital computer code was developed which can be utilized to conduct steady-state performance analyses of the SNAP-8 system. Solutions to problems are obtained by solving a set of non-linear, simultaneous equations. System state-point conditions can be determined for design and off-design operation of the SNAP-8 system and modifications thereof by providing appropriate input data.

The basic computer code was written in a general manner to accomplish the objectives of:

a. Allowing for modifications in performance of individual system components.

b. Having the capability of solving a set of equations for various combinations of variables.

c. Being easily modified to include additional equations or system and component information.

The mathematical approach for solving the set of non-linear equations is described and a section on preparation of input data and the type of output data available is also presented.

A description of the development of the procedure for obtaining influence coefficients is presented. The calculation of influence coefficients is an option which can be specified with input data.

APPROVED:

DEPARTMENT HEAD

W. F. Banks

NOTE: The information in this document is subject to revision as analysis progresses and additional data are acquired.

AEROJET-GENERAL NUCLEONICS
San Ramon, California

EAD - 340

MEMORANDUM

cc: S. Nakazato
H. Snyder
D. Yee
R. Quilici (VKC)
S. Saunders (VKC)
P. Wood (VKC)

TO: R. G. Geimer (12)

FROM: A. B. Burgess

SUBJECT: SCAN, A COMPUTER CODE FOR SNAP-8 SYSTEM ANALYSIS WITH INFLUENCE COEFFICIENT CALCULATION OPTION

I. INTRODUCTION

A previous memo[1] was updated to include:

1) A more detailed exposition of the iterative method employed
2) A description of the influence coefficient calculation option
3) Recent revisions in equations and coefficients

A digital computer code was written to calculate steady state system operating conditions for the SNAP-8 Rankine Cycle. The computer code was written in a general manner to accomplish the objectives of:

1) Allowing for modifications in the performance of individual system components.

2) Having the capability of solving the set of equations for various combinations of the variables.

3) Being easily modified to include additional equations or information.

Component performance data were obtained from TM 4922:65-1-323[2] and supplementary information provided by R. G. Geimer. These data were put in functional form by the method of least squares using the computer codes AGMLR and POLYFIT.

[1] Burgess, A. B., "General Computer Code for SNAP-8 System Analysis", EAD-328, 27 December 1965

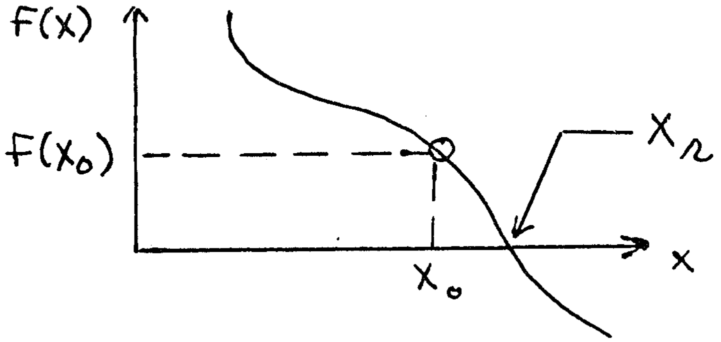
[2] Geimer, R. G., "Analysis of a SNAP-8 EGS Based on Unmodified-1 Component Performance", TM 4922:65-1-323, 3 September 1965

II. COMPUTATIONAL METHOD

A set of 54 functional equations in 69 variables was written to describe the steady state system performance. When 15 independent variables are specified, the set can be solved when the functions are "well-behaved" (i.e., continuous and single-valued).

The resulting set is generally non-linear necessitating an iterative solution. A variation of the Newton-Rapheson method was chosen for the iterative scheme. The modification includes a calculation of the vector norm which furnishes a sufficient condition for convergence of the method and eliminates potential difficulties with divergence.

The Newton method can be most easily understood by considering the determination of the root (or zero value) of a function in one variable, f(x).



Expanding the function in a Taylor series about the point Xo,

$$F(x) = F(x_0) + \left.\frac{dF}{dx}\right|_{x=x_0}(x-x_0) + \ldots + \left.\frac{d^nF}{dx^n}\right|_{x=x_0}\frac{(x-x_0)^n}{n!} + \ldots$$

If we retain only linear terms:

$$F(x) = F(x_0) + \left.\frac{dF}{dx}\right|_{x=x_0}(x-x_0)$$

or $$X = X_o + \frac{F(X) - f(X_o)}{\left.\frac{dF}{dX}\right|_{X = X_o}}$$

Geometrically, the Newton method in one-variable estimates the zero of the function based on an extension of the tangent line. Continued iteration on X gives for the Kth iterate:

$$X^{(K)} = X^{(K-1)} + \frac{f^{(K)} - f^{(K-1)}}{\left. f' \right|_{X = X^{(K-1)}}}$$

The root, Xr, is related to the value of X at each iteration by an error term, ϵ , or:

$$X_r = X^{(o)} + \epsilon^{(o)} = X^{(1)} + \epsilon^{(1)} = \ldots, X^{(K)} + \epsilon^{(K)}$$

Convergence occurs when the value of the function $F^{(k)}$ approaches zero as K increases. As convergence is approached:

$$X^{(K)} \longrightarrow X_r = X^{(K-1)} + \epsilon^{(K-1)}$$

$$\epsilon^{(K)} \longrightarrow 0$$

The one-dimensional Newton algorithm is then:

$$X^{(K)} = X^{(K-1)} - \frac{f^{(K-1)}}{\left. f' \right|_{X = X^{(K-1)}}}$$

or $$X^{(K)} = X^{(K-1)} + \epsilon^{(K-1)}$$

where $$\epsilon^{(k-1)} = -\frac{f^{(k-1)}}{f'\big|_{x=x^{(k-1)}}}$$

The Newton method can be extended to M functions in N variables (where M ≤ N). Retaining only the linear term in the Taylor series expansion, we have:

$$f_i(x_j) = f_i(x_j)^{(0)} + \sum_{j=1}^{N} \frac{\partial f_i}{\partial x_j}\left[x_j - x_j^{(0)}\right] \qquad i = 1 \rightarrow M$$

Representing the entire set compactly as vectors,

$$\vec{F} - \vec{F}^{(0)} = \left[\frac{\partial f_i}{\partial x_j}\right]\left(\vec{X} - \vec{X}^{(0)}\right)$$

where the bracketed term is an MXN matrix of partials. The matrix can be augmented to give a square NXN matrix. This is accomplished by writing the functions Fi , i = M + 1 ——> N as:

$$F_i = X_i - C_k = 0 \; ; \quad i = M+1 \rightarrow N$$

where the C_K's, K = 1 to N-M, are the values of each of the fixed independent variables. The partial matrix is then augmented by inserting a single value of one in the X_i th column, for each of the rows M + 1 through N. The rest of the entries in each of these rows are zeros.

No simple geometrical significance can be attributed to the N-dimensional Newton method.

Continued iteration on the $\vec{x}$ vector gives for the K^{th} iterate:

$$\vec{X}^{(k)} = \vec{X}^{(k-1)} + \frac{\vec{F}^{(k)} - \vec{F}^{(k-1)}}{\left[\frac{\partial f_i}{\partial x_j}\right]_{\vec{X} = \vec{X}^{(k-1)}}}$$

The solution vector $\vec{x}_r$ (i.e., the values of x_j, $j = 1 \longrightarrow N$ which satisfy the given equations) are related by an error vector $\vec{\epsilon}$ to the value of the $\vec{x}$ - vector at each iteration. Thus:

$$\vec{X}_r = \vec{X}^{(0)} + \vec{\epsilon}^{(0)} = \vec{X}^{(1)} + \vec{\epsilon}^{(1)} + , \dots + \vec{X}^{(k)} + \vec{\epsilon}^{(k)}$$

Convergence occurs when the value of each of the functions $F_i^{(k)}$ approaches zero as k increases. As convergence is approached:

$$\vec{X}^{(k)} \longrightarrow \vec{X}_r = \vec{X}^{(k-1)} + \vec{\epsilon}^{(k-1)}$$

$$\vec{\epsilon}^{(k)} \longrightarrow 0$$

The N-dimensional Newton algorithm is then:

$$\vec{X}^{(k)} = \vec{X}^{(k-1)} - \frac{\vec{f}^{(k-1)}}{\left[\frac{\partial F_i}{\partial X_j}\right]_{\vec{X} = \vec{X}^{(k-1)}}}$$

or

$$\vec{X}^{(k)} = \vec{X}^{(k-1)} + \vec{\epsilon}^{(k-1)}$$

where

$$\vec{\epsilon}^{(k-1)} = - \frac{\vec{f}^{(k-1)}}{\left[\frac{\partial F_i}{\partial X_j}\right]_{\vec{X} = \vec{X}^{(k-1)}}} .$$

The $\vec{\epsilon}^{(k-1)}$ vector may be determined by solving the set of linear, simultaneous, algebraic equations represented in matrix form as:

$$\left[\frac{\partial F_i}{\partial X_j}\right]^{(k-1)} \left[\epsilon_j\right]^{(k-1)} = \left[-F_i\right]^{(k-1)}$$

A non-singular solution for the $\overrightarrow{\epsilon}^{(k-1)}$ vector is obtained if the determinant of $\left[\frac{\partial F_i}{\partial X_j}\right]^{(k-1)}$ is not zero.

The procedure followed was to guess the initial solution vector (i.e., the zeroth iterate) and then to evaluate all the functionals, Fi's and the partial derivatives, $\frac{\partial F_i}{\partial X_j}$. The first iterated solution vector is found by solving the resulting set for the $\overrightarrow{\epsilon}^{(0)}$ vector and then adding this vector to the $\overrightarrow{X}^{(0)}$ vector.

$$\overrightarrow{X}^{(1)} = \overrightarrow{X}^{(0)} + \overrightarrow{\epsilon}^{(0)}$$

The procedure is repeated until the desired convergence is achieved.

A sufficient condition for convergence is that the "norm" of the functional vector, f, continually decreases as the iteration proceeds. The sufficient condition can be written as:

$$\left\| f^{(K)} \right\| < \left\| f^{(K-1)} \right\|$$

where the norm of the vector f is defined for convenience as:

$$\left\| f^{(K)} \right\| = \sum_{i=1}^{n} \text{Abs} \left| F_i^{(K)} \right|$$

The matrix of partials was evaluated prior to each iteration by a numerical procedure:

$$\frac{\partial F_i}{\partial X_j}^{(K)} = \frac{F_i^{(K)}(X_1, \ldots X_j + \Delta X_j, \ldots X_n) - F_i^{(K)}(X_1, \ldots X_j, \ldots X_n)}{\Delta X_j^{(K)}}$$

where

$$\Delta X_j^{(K)} = 10^{-4} X_j^{(K)}$$

The Newton method is restricted to well-behaved (i.e., continuous and single valued) functions which are typical of those encountered in engineering problems. The number of iterations required to reach convergence is dependent on the initial guess, i.e., the vector $x^{(o)}$; but the convergence is quadratic for small values of ϵ . 'Modern Numerical Analysis" by R. Glauz of AGC-Sacramento, discusses the Newton method in great detail.

III. PREPARATION OF INPUT DATA

The first card is a title card containing alphanumeric symbols in columns #2 through 72. Column #1 of the title card of the first problem in a sequence of problems must be left blank. If it is desired to run a series of problems (i.e., stacked problems) based on the same performance coefficients, an "R" (for repeat) is punched in column #1 of the second and all subsequent title cards.

The next cards (see Figure 4 for sample input sheets) read in the performance coefficients. The current format requires seventeen cards to read in all of the performance coefficients. These cards are followed by the control card containing the parameters:

NVAR - the number of variables, N

NFNS - the number of fuctionals, M

KMAX - the maximum number of iterations allowed

ER - the maximum residual of any functional (i.e., the convergence criterion) ER = 10^{-3} should be sufficient

APRNT - an option parameter: if APRNT > 0 the final values, corresponding to the converged X-vector, of the influence coefficients are printed out. Only the non-zero influence coefficients and their identification numbers are printed.

The control card is followed by an initial guess on each of the variables. These are punched eight to a card and so require nine cards for the case of 69 total variables.

The last card consists of the identification number of the fixed independent variables arranged in ascending numerical order (e.g., 3, 4, 5, 6, 25, etc.). The number of these fixed variables should be NVAR-NFNS.

The input format is:

Card	Format	No. of Cards
Title	12A6	1
Performance of coefficients	8F9.0	17
Control	3I3, 2F9.0	1
Initial guess	8F9.0	9
Fixed independent variables	16I3	1
		29 Total

When running stacked problems, the performance coefficients are read in only with the first problem. Subsequent problems then consist of: a title card, the control card, initial guess on all the variables, and the identification numbers of the fixed variables.

IV. COMPUTER OUTPUT

The output from each problem consists of:

1) The information on the title card
2) Print out of the input quantities (performance coefficients, control parameters, initial guesses, and identification numbers of fixed independent variables)
3) The values of the non-zero influence coefficients and their identification numbers (if APRNT > 0 on the control card)
4) A diagram of the system with values of temperatures, pressures, flow rates, heat transfer rates and output power included
5) The values of a number of calculated quantities, e.g., KVA, cycle efficiency, turbine efficiency, etc.
6) The final values of the variables. These columns of X's are headed by the last values of the:

   iteration number, K

   the norm, TNRM

   the maximum residual of any of the functionals, RFMAX
7) The values of the norm after each iteration to show the rate of convergence of the problem.

There are four possible error messages. These are:

1) The repeat option on the title card is incorrectly specified. The message is "repeat option on title card incorrectly specified".
2) The number of iterations exceed KMAX without convergence being attained. The message is "iterations loop exceeds KMAX".
3) The number of consecutive inner loop iterations (i.e., halving of the ϵ_i's to insure a sufficient condition for convergence) exceeds five. The message is "halving loop exceeds 5".
4) The matrix of partials used by the SOLF4 subroutine to calculate the values of the ϵ_i's or to calculate the influence coefficients is singular (i.e., one of the rows or columns of the matrix contains only zero entries and a solution is not obtainable). The message is "equation set singular in solve."

V. DETAILS OF THE COMPUTER CODE

The computer code package consists of the main program (SCAN) and four subroutines (RFSUB, PARTL, RFNORM, and SOLF4).

The following sequence (as shown in the flow chart of Figure 3) is followed in the computer program:

(1) Read in and write out the input data.

(2) Initialize the arrays to zero.

(3) Using the initial $\vec{x}$ vector calculate the $-f_i$'s, their norm, and the maximum residual of any f_i. (The RF(I)'s, TNRM, and RFMAX)

(4) Test for convergence if converged go to step (10) if not converged go to step (5). (Test RFMAX vs ER)

(5) Calculate the matrix of partials. (The A(I, J)'s)

(6) Solve for the ϵ_j's. (The EPS(I)'s)

(7) Reset the matrix of partials and the $\vec{f}$ vector to zero.

(8) Calculate a new trial solution vector $\vec{x}$. (The TX(I)'s)

(9) Calculate the new $-f_i$'s, a new trial norm and the maximum residual of any f_i. If the new trial norm is less than the old norm, the trial solution vector is accepted and the outer iteration loop repeated by returning to step (4).

If the new trial norm is greater than the old norm, the ϵ_j's are halved and the inner iteration loop repeated by returning to step (8).

(10) Test the influence coefficient option (APRNT). If the influence coefficients are not required, a series of non-iterative calculations are completed, a system map is printed out, and the values of the converged X's printed. If the influence coefficients are required, the partial derivatives are calculated using the converged X-vector and a series of sets of simultaneous equations are solved to give the influence coefficients. These are printed prior to printing the system map and the converged X's.

A description of each of the subroutines follows:

1) RFSUB - This routine lists the functional equations of the set to be solved (a total of NFNS equations). The quantities RF(I) in this routine when called by the main program are equal to the negative of each functional, i.e., $-f_i$, and can be used directly as the right hand side of the matrix. The RFSUB routine is also used in the numerical calculation of the partials and is called by the subroutine PARTL. When used in the PARTL routine, the parameter OPTION in the RFSUB call list is set equal to 2, and the quantity IROW in the call list equals the number of the functional whose partial is being evaluated. In this case, the RFSUB routine calculates the quantity TERM which is:

$$f_i(x_1, x_2, \ldots . x_u + \Delta x_j, \ldots . x_n)_K \qquad i = \text{IROW}$$

In the main program the parameter OPTION in the RFSUB call list is set equal to 1.0 and the quantity RF(I) = $f_i^{(K)}$ $(x_1, x_2, \ldots . x_j, \ldots . x_n)$ is calculated for I equal 1 to n.

2) RFNORM - This simple routine sums the absolute values of the RF(I)'s and calculates the absolute value of the largest RF(I). These quantities, output as TNRM and RFMAX, respectively, are used in the "norm" and convergence tests.

3) PARTL - This routine numerically evaluates the matrix of partials during each iteration. The data key used in this routine corresponds to the variables present in the functional equations. The quantity KEY (K_1, K_2) identifies the non-zero partials by sequentially listing the number of each function as K_1 and the identification number of each variable appearing in that function as K_2. The dimension of K_1 is NFNS, the number of functionals. The dimension of K_2 is the maximum number of variables appearing in any functional (14 for this set of equations). When the number of variables in a functional is less than the maximum, the appropriate number of zeros must be inserted.

The matrix of partials, A (I, J), is initially set equal to zero. When the PARTL routine is called, the data key is scanned until a value of KEY (I, J) greater than zero is reached. The index J is set equal to the value of KEY (i.e., the identification number of the variable) and the variable X(J) is set aside and stored as the quantity XSAVE. The variable X(J) is then incremented by DX, where DX is one ten-thousandths of X(J). The RFSUB routine is used to initiate the numerical evaluation of the partial. The partial is then

$$\text{PART (I,J)} = [-\text{ TERM} + \text{RF(I)}]/\text{DX}$$

For each of the bottom rows of the matrix, i.e., rows (NVAR-NFNS + 1) to NVAR, it is necessary to insert a value of one in the column corresponding to the identification number of that particular variable read in as a fixed number. These variables are the input quantities COL(I). COL(I) is then used in PARTL to identify the appropriate column number. Each row in the lower part of the matrix will contain zeros except for a single value of one.

4) SOLF4 - This routine solves a set of linear, simultaneous algebraic equations by the method of Gaussian elimination. Normalization and pivotal condensation are used to minimize round-off error. The routine is used to solve the iterative correction vector, EPS(I), using the matrix of partials A(I,J) and the negative of the functionals, RF(I). The routine is also used to solve for the influence coefficients, CF(I,N2) where the N2's refer to the identification numbers of the fixed independent variables.

At each stage of the elimination, each row is divided by its coefficient of largest absolute value. The equation having the largest coefficient in the last column (i.e., J = Nth column) is then placed at the bottom of the set. All other equations are combined with it to eliminate the coefficients of the last variable, ϵ_N, forming a set of N-1 equations in the remaining variables. The process is repeated until a triangular set is formed and the back solution is then performed to calculate each of the ϵ_i's.

The quantity ERR was included in the SOLF4 call list to indicate a singular matrix. In this event an error message is written and the iterative solution terminated.

VI. DEVELOPMENT OF PROCEDURE FOR CALCULATING INFLUENCE COEFFICIENTS

Assume that N expressions for the N functionals, Fi's ,are available where:

$$F_i(x_1,x_2,\dots x_j,\dots x_N,x_{N+1},\dots x_{N+K},\dots x_{N+M}) = 0 \text{ , } i = 1 \rightarrow N$$

and X_j,j = 1 → N are the dependent variables
X_{n+k} k = 1—> M are the fixed independent variables.

(In the FORTRAN terminology M = NXTRA, N = NFNS, and M + N = NVAR). The problem to be solved is the calculation of the influence coefficients, $\frac{\partial X_j}{\partial X_{N+K}}$. The total number of influence coefficients will be N x M, but many of these are zeros. For the SNAP-8 system equations, N = 54 and M = 15; so there are 810 possible coefficients describing the system.

Applying the definition of the total derivative, we can write:

$$\sum_{j=1}^{N} \frac{\partial F_i}{\partial X_j} dX_j + \sum_{k=1}^{M} \frac{\partial F_i}{\partial X_{N+k}} dX_{N+k} = 0 \quad , \quad i = 1 \rightarrow N$$

or in matrix form:

$$\left[\frac{\partial F_i}{\partial X_j}\right]\left[dX_j\right] = -\left[\sum_{k=1}^{M} \frac{\partial F_i}{\partial X_{N+k}} dX_{N+k}\right]$$

Applying Cramer's rule gives a theoretical solution for the set of dx_j's, if the determinant of the partial matrix is not zero. If we let:

$$D = \det\left[\frac{\partial F_i}{\partial X_j}\right]$$

$D_K = \det\left[\frac{\partial F_i}{\partial X_j}\right]$ with the jth column replaced by the column $\sum_{k=1}^{M} \frac{\partial F_i}{\partial X_{N+k}}$

then $dX_j = -\sum_{k=1}^{M} D_k \, dX_{N+k} / D$

Since $dX_j = \sum_{k=1}^{M} \frac{\partial X_j}{\partial X_{N+k}} dX_{N+k}$

Equating the dX_j's gives:

$$\frac{\partial X_j}{\partial X_{N+K}} = -\frac{D_K}{D} \qquad \begin{array}{l} j = 1 \rightarrow N \\ K = 1 \rightarrow M \end{array}$$

The evaluation of all the required determanant makes the Cramer's rule method too lengthy to be practical.

An alternate procedure is based on inverting the partial matrix.

Let A = partial matrix with elements a_i,j

V = A^{-1} with elements v_i,j

Then $$\left[dx_j\right] = \left[V_{j,i}\right]\left[-\sum_{k=1}^{M}\frac{\partial f_i}{\partial x_{N+k}}dx_{N+k}\right]$$

Or written out as a matrix multiplication:

$$dx_j = \sum_{i=1}^{N}(v_{j,i})\left(-\sum_{k=1}^{M}\frac{\partial f_i}{\partial x_{N+k}}dx_{N+k}\right), \quad j=1\to N$$

we can also write:

$$dx_j = \sum_{k=1}^{M}\frac{\partial x_j}{\partial x_{N+k}}dx_{N+k} \qquad j=1\to N$$

so $$\frac{\partial x_j}{\partial x_{N+k}} = \sum_{i=1}^{N}(v_{j,i})\left(-\frac{\partial f_i}{\partial x_{N+k}}\right) \qquad \begin{matrix} j=1\to N \\ k=1\to M \end{matrix}$$

This procedure was coded using a matrix inversion subroutine from the Azusa program library (Job #312 by J. Schweiter) based on the method of rank annihilation. Correct values for influence coefficients were obtained for a small sample system, but for the larger system of equations describing SNAP-8 unreasonable answers were obtained. The poor answers probably resulted from accumulated inaccuracies in the inversion routine.

An alternate approach was coded in which the influence coefficients were obtained by solving a linear set of N equations M times

Let $F_i(X_1,X_2,\ldots X_j,\ldots .X_N,X_{N+1},\ldots .X_{N+K},\ldots .X_{N+M}) = 0 \quad i = 1 \to N.$

Differentiating the functionals with respect to each of the fixed independent variables, we obtain:

$$i=1\to N, \qquad \sum_{j=1}^{N}\frac{\partial f_i}{\partial x_j}\frac{\partial x_j}{\partial x_{N+k}} + \frac{\partial f_i}{\partial x_{N+k}} = 0$$

or in matrix form:

$$\left[\frac{\partial f_i}{\partial x_j}\right]\left[\frac{\partial x_j}{\partial x_{N+k}}\right] = \left[-\frac{\partial f_i}{\partial x_{N+k}}\right]$$

When the control card parameter, APRNT, is set equal to a positive quantity in the input data, the influence coefficients are calculated upon convergence of the Newton iterative method. The solution vector (the X_j's) is used by the PARTL subroutine to calculate all the $\frac{\partial F_i}{\partial X_j}$'s. The SOLF4 subroutine is then used M times with each of the M column vectors - $\frac{\partial F_i}{\partial X_{N+K}}$ substituted in the right side and solved for the $\frac{\partial X_j}{\partial X_{N+K}}$'s. Each of the non-zero influence coefficients is printed out with its row and column identification.

VII. DEVELOPMENT OF EQUATIONS FOR SYSTEM ANALYSIS

The discussion of the equation development is divided into four general categories:

A. Pump Motor Assemblies (PMA's) and Loop Pressure Drops

For the primary, mercury, heat rejection and lube coolant loops, pump performance curves were fit to the least squares criterion as a function of one variable using the AGMLR code. The performance curves were taken from R. Geimer's compilation$^{(2)}$ and expressed in the following units:

Q = volumetric flow in gpm

W = mass flow in lb/hr

ΔH = head in ft

q = PMA input power in Kw

PF = power factor, n.d.

For the primary loop (indicated by the symbol and subscript N) we have:

$$(1) \quad Q_N = W_N/N_1 \quad (N_1 = 367, \text{ p. 10, Ref. 2})$$

$$(2) \quad \Delta H_N = N_2 + N_3 Q_N - N_4 Q_N^2 \quad (\text{Fig. A-7, Ref. 2})$$

$$(3) \quad q_n = N_5 + N_6 Q_N - N_7 Q_N^2 \quad (\text{Fig. A-7, Ref. 2})$$

$$(4) \quad PF_N = N_8 + N_9 q_N - N_{10} q_N^2 \quad (\text{Fig. A - 8, Ref. 2})$$

The signs of the performance coefficients (the N's) were chosen so that they may be input as positive quantities and the sign ignored.

$$(5) \quad N_{16} \Delta H_N = \left\{ N_{11} + N_{12} + N_{13} + N_{14} + C_N \right\} \left(\frac{W_N}{41300} \right)^2 + N_{15} \left(\frac{W_N}{48100} \right)^2$$

where N_{11}, N_{12}, and N_{13} express piping pressure loss coefficients between the points H to A, C to D, and F to G respectively on Figure 1. N_{14}, N_{15}, and C_N represent pressure loss coefficients across the reactor, boiler, and trim orifice respectively. N_{16} is the pump pressure rise coefficient

(Ref. 2, p.10). Similar expressions were obtained for the mercury loop (subscript H) as:

$$(6)\quad Q_H = W_H/H_1 \quad (H_1 = 6480, \text{p. 11, Ref. 2})$$

$$(7)\quad \Delta H_H = H_2 - H_3 Q_H - H_4 Q_H^2 \quad (\text{Fig. A-10, Ref. 2})$$

$$(8)\quad q_H = H_5 + H_6 Q_H + H_7 Q_H^2 \quad (\text{Fig. A-10, Ref. 2})$$

$$(9)\quad PF_H = H_8 + H_9 q_H - H_{10} q_H^2 \quad (\text{Fig. A-11, Ref. 2})$$

$$(10)\quad H_{16}\Delta H_H = \left\{ \frac{H_{11} + C_H}{(11750)^2} + \frac{H_{12}}{(10000)^2} \right\} W_{HT}^2 + H_{18}\Delta T_P + \frac{\sqrt{T_4 + 460}}{H_{13}} (1-F)\ W_{HT} - P_5 + H_{14} + H_{15}$$

In equation (10) the H coefficients represent:

H_{11} pipe pressure loss coefficient from boiler to turbine, psi

H_{12} boiler pressure loss coefficient, psi

H_{13} turbine nozzle constant (p.7, Ref. 2), $\text{lb/hr-}^{\circ}R^{\frac{1}{2}}\text{/psi}$

H_{14} pipe pressure loss from turbine to condenser, psi

H_{15} condenser pressure loss, psi

H_{16} pump pressure rise coefficient, psi/ft head

H_{18} empirical boiler pressure loss coefficient, psi/°F

C_H trim orifice pressure loss coefficient, psi

The other quantities in Equation (10) are:

T_4 turbine inlet temperature, °F

F liquid carryover fraction, n.d.

W_{HT} total mercury flow rate (vapor plus liquid), lb/hr

ΔT_P boiler pinch point temperature difference = $T_B - T_2$ (Fig.1) °F

P_5 turbine exhaust pressure, psia

Similar expressions for the heat rejection loop (subscript R) are:

(11) $Q_R = W_R/C_1$ (C_1 = 407, p. 11, Ref. 2)

(12) $\Delta H_R = C_2 + C_3 Q_R - C_4 Q_R^2$ (Fig. A-9 , Ref. 2)

(13) $q_R = C_5 + C_6 Q_R - C_7 Q_R^2$ (Fig. A-9, Ref. 2)

(14) $PF_R = C_8 + C_9 q_R - C_{10} q_R^2$ (Fig. A-8, Ref. 2)

(15) $$C_{16} \Delta H_R = \left\{ \frac{C_{11}}{(39500)^2} + \frac{C_{12}+C_{13}+C_{14}-C_R}{(38100)^2} + \frac{C_{15}}{(39300)^2} \frac{(130)^2}{(NT)^2} \right\} W_R^2$$

The C coefficients in equation (15) represent:

C_{11} condenser pressure loss coefficient, psi

C_{12} pipe pressure loss coefficient condenser to radiator, psi

C_{13} pipe pressure loss coefficient radiator to trim orifice, psi

C_{14} pipe pressure loss coefficient trim orifice to condenser

C_{15} radiator pressure loss coefficient, psi

C_{16} pump pressure rise coefficient, psi/°F

C_R trim orifice pressure loss coefficient, psi

NT number of radiator flow tubes (equal flow and pressure drop assumed in each tube), n.d.

The analysis of the lube-cooler loop (subscript LC) is restricted to determining pump power requirements using the following equations:

(16) $Q_{LC} = W_{LC}/L_1$

(17) $q_{LC} = L_2 + L_3 Q_{LC} + L_4 Q_{LC}^2$ (Fig. A-12, Ref. 2)

(18) $PF_{LC} = L_5 + L_6 Q_{LC} - L_7 Q_{LC}^2$ (Fig. A-13, Ref. 2)

Generally the value of W_{LC} will be a fixed input quantity.

B. Turbo-Alternator and Loads

A schematic of the energy input and distribution of the output loads is shown in Figure 2. The alternator output power, q_{TRM}, is divided between:

1) vehicle load
2) power required to run cycle
3) excess power dumped back into primary loop

The power dumped back into the primary loop, q_{PLR}, includes:

q_{PLM}	minimum parasitic load
q_S	power required for speed control systems stability
q_{ex}	power in excess of vehicle load requirements (if any)

The power dissipated in running the cycle includes the input power to each of the loop pumps (q_N, q_H, q_R and q_{LC}) plus the power dissipated in the control system, q_{SC}.

There are two different possibilities for the vehicle load:

1) The vehicle load is specified as a fixed input quanaity, q_{LDS}, and the excess power output, q_{ex}, is dumped into the primary loop.

2) The vehicle load is not specified and is calculated from the cycle state points and performance curves. This case is obtained by setting $q_{ex} = 0$. The relation between specified load, q_{LDS}, and actual load, q_{LD}, is:

$$(19) \quad q_{ex} = q_{LD} - q_{LDS}$$

The expressions for q_{PLR} and Q_{TRM} are:

$$(20) \quad q_{PLR} = q_{PLM} + q_s + q_{ex}$$

$$(21) \quad q_{TRM} = q_N + q_H + q_R + q_{LC} + q_{SC} + q_{LD} + q_{PLM} + q_S$$

Expressions for the loop pump power factors were obtained from the previously mentioned performance curves of Reference 2. The power factor, PF_x, for the speed control system was obtained as a function of load from Figure A-16, Ref. 2, using the AGMLR data fitting code:

$$(22)\quad PF_x = G_1 + G_2(q_{SC} + q_{PLR}) - G_3\ (q_{SC} + q_{PLR})^2$$

where the G's are performance coefficients (input quantities).

Then the expressions for total KVA and overall alternator power factor, PF_a, are:

$$(23)\quad KVA = \sqrt{(q_N + q_H + q_R + q_{LC} + q_{SC} + q_{PLR} + q_{LDS})^2 + \left(\frac{q_N}{PF_N}\sqrt{1-PF_N^2} + \frac{q_H}{PF_H}\sqrt{1-PF_H^2} + \ldots . + \frac{q_{LDS}}{PF_{LO}}\sqrt{1-PF_{LD}^2}\right)^2}$$

$$(24)\quad PF_A = q_{TRM}/KVA$$

where PF_{LD} is the power factor for the vehicle load.

An expression for the alternator efficiency, η_A, was obtained from performance curve A-5, Reference 2, using the POLYFIT code:

$$(25)\quad \eta_A = G_4 + G_5Z_1 - G_7Z_1^2 + G_8Z_2 + G_{10}Z_1Z_2 - G_{11}Z_2^2$$

where

$$Z_1 = PF_A - G_6$$

$$Z_2 = q_{TRM} - G_9$$

Using the definition of alternator efficiency:

$$(26)\quad q_T = Q_{BS} + q_{TRM}/\eta_A$$

where q_T is the turbine output power

q_{BS} is the turbine bearing and seal losses

C. Expansion in the Turbine

The expression for the turbine efficiency, η_T, was obtained from the performance curve A-4 (revised) Reference 2, using POLYFIT to give:

$$(27)\quad \eta_T = G_{14} - G_{15}Z_1 + G_{16}Z_1^2 + G_{17}Z_2 - G_{18}Z_1Z_2 +$$

$$G_{19}\, Z_1^2 Z_2 - G_{20}Z_2^2 + G_{21}Z_1Z_2^2 - G_{22}Z_1^2Z_2^2$$

where

$$Z_1 = 100F - G_{12}$$

$$Z_2 = U/C_o - G_{13}$$

U = turbine peripheral speed = .0223 N

N = rpm = 12,000

C_o = theoretical spouting velocity = $\sqrt{2\, g_C\, \Delta h_{isen}}$

$$(u/C_o) = \frac{267.5}{\sqrt{2g_c\ \Delta h_{isen}}} = \frac{1.196}{\sqrt{h_4 - h_{5isen}}}$$

h_4 is the enthalpy (Btu/lb) at the turbine inlet and h_{5isen} is the enthalpy at the turbine exhaust for an isentropic expansion.

The vapor region enthalpy for mercury (ignoring pressure dependency) can be written as a function of temperature:

$$(28)\quad h_4 = 130.25 + CP_vT_4$$

The entropy of an ideal gas is:

$$S_2 = S_1 + R\, \ell n\left(\frac{P_1}{P_2}\right) - CP_v\, \ell n\left(\frac{T_1}{T_2}\right)$$

Using the ideal gas form and data from the Mollier diagram taken at 1100 and 1400°F and 14 and 280 psia, the expression for the entropy at the turbine inlet is:

$$(29)\quad S_4 = .1177 - .0100\, \ell n\left(\frac{P_4}{280}\right) + .0250\, \ell n\left(\frac{T_4 + 460}{1560}\right)$$

with S in Btu/lb-°R

From the definition of turbine efficiency:

$$(30)\quad h_5 = h_4 - \eta_T \ (h_4 - h_{5\ is})$$

where h_5 is the turbine exhaust enthalpy. For a real (i.e., non-isentropic) expansion. The turbine output power is:

$$(31)\quad h_{5\ is} = h_4 - 3413 \ \frac{q_T}{\eta_T W_H \ (1-F)}$$

The state of the expanded mercury in the condensed region (under the dome) can be expressed as a function of two thermodynamic variables. Writing P = Fn (h, s), the region of the Mollier chart between 85 and 100% quality and between 8 and 30 psia was fitted using POLYFIT to give:

$$(32)\quad P_5 = G_{25} + G_{26}Z_1 + G_{27}Z_1^2 - G_{28}Z_2 - G_{29}Z_1Z_2 + G_{30}Z_1^2Z_2 + G_{31}Z_2^2 - G_{32}Z_1Z_2^2 - G_{33}Z_1^2Z_2^2$$

where

$$Z_1 = h_{5is} - G_{34}$$

$$Z_2 = S_4 - G_{35} \qquad \text{where} \quad S_4 = S_{5is}$$

$$P_5 = P_{5is}$$

The pressure drop from the turbine exhaust to the condenser inlet is:

$$(33)\quad P_5 - P_6 = H_{14}$$

The mercury condenses at constant temperature T_6 until the saturated liquid enthalpy, h_{6SL}, is reached and subcooling begins. The saturated liquid enthalpy is a function of one thermodynamic variable:

$$(34)\quad h_{6SL} = 3.286 \ \ln P_6 + 12.14 \ \text{for} \ 8 \le P_6 \le 30$$

For pressures outside of this range a third or fourth order polynomial will give sufficient accuracy. Finally for choked flow through the first stage nozzle of the turbine (p.7, Ref. 2):

$$(35)\quad P_4 = \sqrt{\frac{T_4 + 460}{H_{13}}}\ (1-F)\ W_{HT}$$

4) Energy Transfer Between Components and Linking Equations

For the primary loop the overall energy balance is:

$$q_{PLR} + q_{RX} = q_{LN} + q_{BV} + q_{BL}$$

where

q_{RX} = reactor power, kw

q_{LN} = power loss in pipe between the reactor and boiler, kw

q_{BV} = power transferred across the boiler in the superheat and latent heat regions, kw

q_{BL} = power transferred across the boiler in the preheat region, kw.

Writing this as five separate equations where the temperature points are shown on Figure 1 , we have:

$$(36)\quad q_{PLR} = W_N CP_N \ (T_G - T_C)/3413$$

$$(37)\quad q_{RX} = W_N CP_N \ (T_H - T_G)/3414$$

$$(38)\quad q_{LN} = W_N CP_N \ (T_H - T_A)/3414$$

$$(39)\quad q_{BV} = W_N CP_N \ (T_A - T_B)/3414$$

$$(40)\quad q_{BL} = W_N CP_N \ (T_B - T_C)/3414$$

Using the pinch point definition to connect the primary and the mercury loops:

$$(41)\quad \Delta T_p = T_B - T_2$$

Combining the mercury side boiler pressure drop and boiler-turbine pipe pressure loss equations (pp 5-6, Ref. 2) we have:

$$(42)\quad (P_2 - P_4) = \frac{W_H^{\ 2}}{(10000)^2} \left\{ .724H_{11} + H_{12} - .756H_{17} \right\} + H_{18}\Delta T_P$$

The H coefficients in equation (42) represent:

H11, H12, H18, and ΔT_p are defined for equation (10)

H17 - pressure loss coefficient from boiler inlet to saturation point (i.e. pinch point).

The portion of the power from the mercury pump that is transmitted to the mercury, q_p, was obtained by fitting performance curve A-16, Reference 2:

$$(43)\quad q_p = G_{23} + G_{24}Q_H$$

Writing the energy balance equations between temperature points as shown in Figure 1, we have:

$$(44)\quad q_P = W_H CP_L (T_1 - T_7)/3414$$

$$(45)\quad q_{BL} = W_H CP_L (T_2 - T_1)/3413$$

$$(46)\quad q_{BV} = W_H(1-F) \left[\Delta h_{fg}(T_2) + CP_V (T_3 - T_2) \right]/3413$$

where Δh_{fg} $(T_2) = 132.15 - .00825\ T_2$ = latent heat of vaporization Btu/lb

$$(47)\quad q_{RAD} = W_H \left[(1-F)\ (h_5 - h_{6SL}) + (h_{6SL} - CP_L T_7) \right]/3413$$

$$(48)\quad q_{LH} = W_H CP_V (T_3 - T_4)/3413$$

where q_{RAD} = power transferred across the condenser to the heat rejection loop and rejected to space, kw

q_{LH} = power loss in pipe between the boiler and the turbine, kw

Empirical relations describing the boiler and condenser performance are used to link the mercury loop to the primary and heat rejection loops. Using the notation of Figure 1:

$$(49)\quad T_J = T_6 - 10.$$

where

$$T_6 = \frac{G_{36}}{G_{37} - \ell n\, P_6} - 460 = \text{saturation temp, } {}^{o}F$$

$$(50)\quad T_A = T_3 + 20$$

The overall energy balance for the heat rejection loop is:

$$(51)\quad q_{RAD} = W_R CP_R \; (T_J - T_I)/3413$$

POLYFIT was used to fit the radiator heat rejection performance curves (Figure A-6 and A-6' of Reference 2 in the form:

$$(52)\quad qR^* = \frac{q_{RAD}}{NT} = \Big[R_1 + R_2 Z_2 - R_3 Z_2^2 + R_4 Z_1 + R_5 Z_1 Z_2 - R_6 Z_1 Z_2^2 - R_7 Z_1^2 - R_8 Z_1^2 Z_2 - R_9 Z_1^2 \; Z_2^2\Big]$$

where

$$Z_1 = T_I - R_{10}$$

$$Z_2 = T_J - R_{11}$$

A different set of the R coefficients are used for each of the radiator performance curves to represent the separate curves for the sun and shade cases.

Using an empirical expression for the condenser terminal temperature difference:

$$(53)\quad T_I = T_7 - 2$$

Finally, for the saturated mercury at the pinch point:

$$(54)\quad T_2 = \frac{G_{36}}{G_{37} - \ell n\, P_2}$$

When the iterative solution of the non-linear set of equations has converged, additional non-iterative calculations are performed to obtain values of temperature and pressure at intermediate points around the various loops. These values are then displayed as output on a schematic cycle diagram drawn by the computer.

Using Figure A-9 of Reference 2, the curve of required NPSH at the pump inlet was fitted with AGMLR. The pump inlet pressure of the primary loop, P_D, is

$$(55)\quad P_D = \left[NP_1 - NP_2 Q_N + NP_3 Q_N^{\ 2} \right] N_{16}$$

where NP_1, NP_2 and NP_3 are the coefficients obtained from No. A-9, Ref. 2. Having determined P_D, the calculations are continued around the loop to the reactor outlet:

$$(56)\quad P_E = P_D + N_{16}\Delta H_N \qquad\qquad N_{16} = .3175$$

$$(57)\quad P_F = P_E - C_N\ (W_N/41300)^2$$

$$(58)\quad P_G = P_F - N_{13}\ (W_N/41300)^2 \qquad N_{13} = 6.2$$

$$(59)\quad P_H = P_G - N_{14}\ (W_N/41300)^2 \qquad N_{14} = 4.3$$

The reactor outlet pressure, P_H, is tested to see if the minimum pressure criterion (Reference 2) of 35 psi is met. If not, P_H is set to:

$$(59a)\quad P_H = 35 + PSFN$$

where PSFN is a safety factor (an input quantity typically 3.psi) and equations (58) through (55) are recomputed in reverse order.

If the 35 psi minimum pressure criterion is satisfied, the primary loop pressure calculations are continued:

(60) $P_A = P_H - N_{11}\ (W_H/41300)^2 \qquad N_{11} = 2.2$

(61) $P_C = P_A - N_{15}\ (W_N/48100)^2 \qquad N_{15} = 1.5$

The pump inlet pressure of the heat rejection loop, P_M, is:

(62) $P_M = \left[NP_4 - NP_5 Q_R + NP_6 Q_R^2 \right] C_{16} + PSFR$

where PSFR is a safety factor (an input quantity typically 5 psi) and NP_4, NP_5 and NP_6 are available for use with a curve fit (currently they are set equal to the same values as NP_1, NP_2 and NP_3, respectively. Having determined P_M, the calculations are continued around the loop:

(63) $P_N = P_M + C_{16}\ \Delta H_R \qquad C_{16} = .3525$

(64) $P_O = P_N - C_R\ (W_R/38100)^2$

(65) $P_I = P_O - C_{14} \qquad C_{14} = 2.6$

(66) $P_J = P_I - C_{11}\ (W_R/39500)^2 \qquad C_{11} = 4.0$

(67) $P_K = P_J - C_{12}\ (W_R/38100)^2 \qquad C_{12} = 4.6$

(68) $P_L = P_K - C_{15}\ (W_R/39300)^2\ (130/NT)^2 \qquad C_{15} = 18.8$

For the mercury loop:

(69) $P_3 = P_4 + H_{11}\ (W_H/11750)^2 \qquad H_{11} = 10.0$

(70) $P_1 = P_3 + H_{12}\ (WHT/10000)^2 + H_{18}\ \Delta T_P$

(71) $P_7 = P_6 - H_{15}$

(72) $$T_5 = \frac{G_{36}}{G_{37} - \ln P_5} - 460$$

(73) $$T_6 = \frac{G_{36}}{G_{37} - \ln P_6} - 460$$

(74) $P_9 = P_8 + H_{16} \Delta H_{HT}$

It is also assumed that

$P_8 = P_7$

$T_K = T_J$

$T_D = T_E = T_C$

$T_L = T_M = T_N = T_I$

$T_8 = T_7$

$T_9 = T_1$

Finally, the overall cycle efficiency, CEF, and the net turbine work, QNET, are calculated:

(75) CEF = 100 QLDS/QRX

(76) QNET = QT - QBS

Figure 3
COMPUTER CODE FLOW CHART

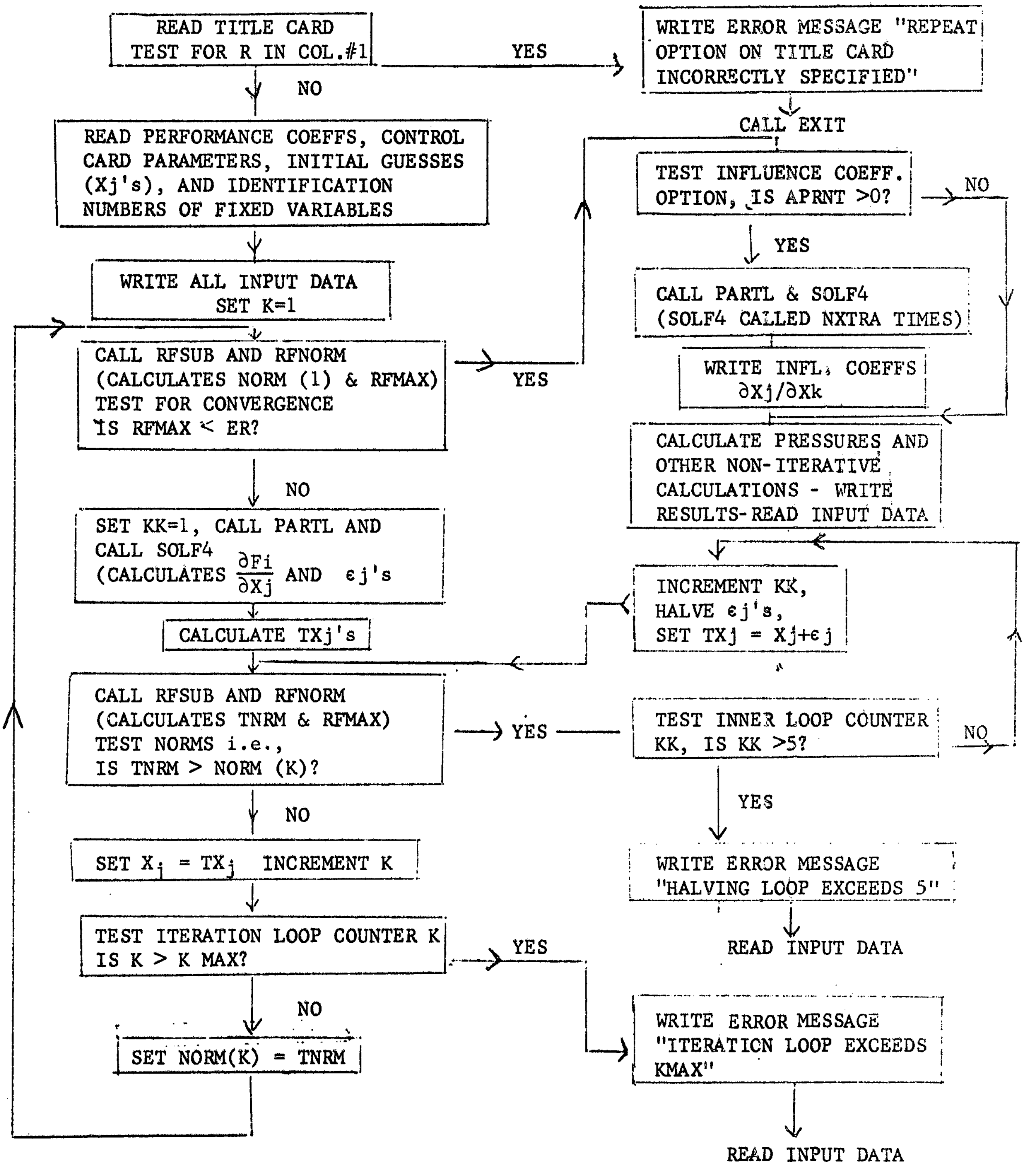

READ TITLE CARD
TEST FOR R IN COL.#1
YES
WRITE ERROR MESSAGE "REPEAT OPTION ON TITLE CARD INCORRECTLY SPECIFIED"
CALL EXIT
NO
READ PERFORMANCE COEFFS, CONTROL CARD PARAMETERS, INITIAL GUESSES (Xj's), AND IDENTIFICATION NUMBERS OF FIXED VARIABLES
WRITE ALL INPUT DATA
SET K=1
TEST INFLUENCE COEFF. OPTION, IS APRNT >0?
NO
YES
CALL PARTL & SOLF4 (SOLF4 CALLED NXTRA TIMES)
WRITE INFL. COEFFS ∂Xj/∂Xk
CALL RFSUB AND RFNORM (CALCULATES NORM (1) & RFMAX) TEST FOR CONVERGENCE IS RFMAX < ER?
YES
CALCULATE PRESSURES AND OTHER NON-ITERATIVE CALCULATIONS - WRITE RESULTS-READ INPUT DATA
NO
SET KK=1, CALL PARTL AND CALL SOLF4 (CALCULATES ∂Fi/∂Xj AND εj's
INCREMENT KK, HALVE εj's, SET TXj = Xj+εj
CALCULATE TXj's
CALL RFSUB AND RFNORM (CALCULATES TNRM & RFMAX) TEST NORMS i.e., IS TNRM > NORM (K)?
YES
TEST INNER LOOP COUNTER KK, IS KK >5?
NO
NO
YES
SET Xj = TXj INCREMENT K
WRITE ERROR MESSAGE "HALVING LOOP EXCEEDS 5"
READ INPUT DATA
TEST ITERATION LOOP COUNTER K IS K > K MAX?
YES
NO
WRITE ERROR MESSAGE "ITERATION LOOP EXCEEDS KMAX"
SET NORM(K) = TNRM
READ INPUT DATA

FIGURE 1
LØØP DIAGRAM

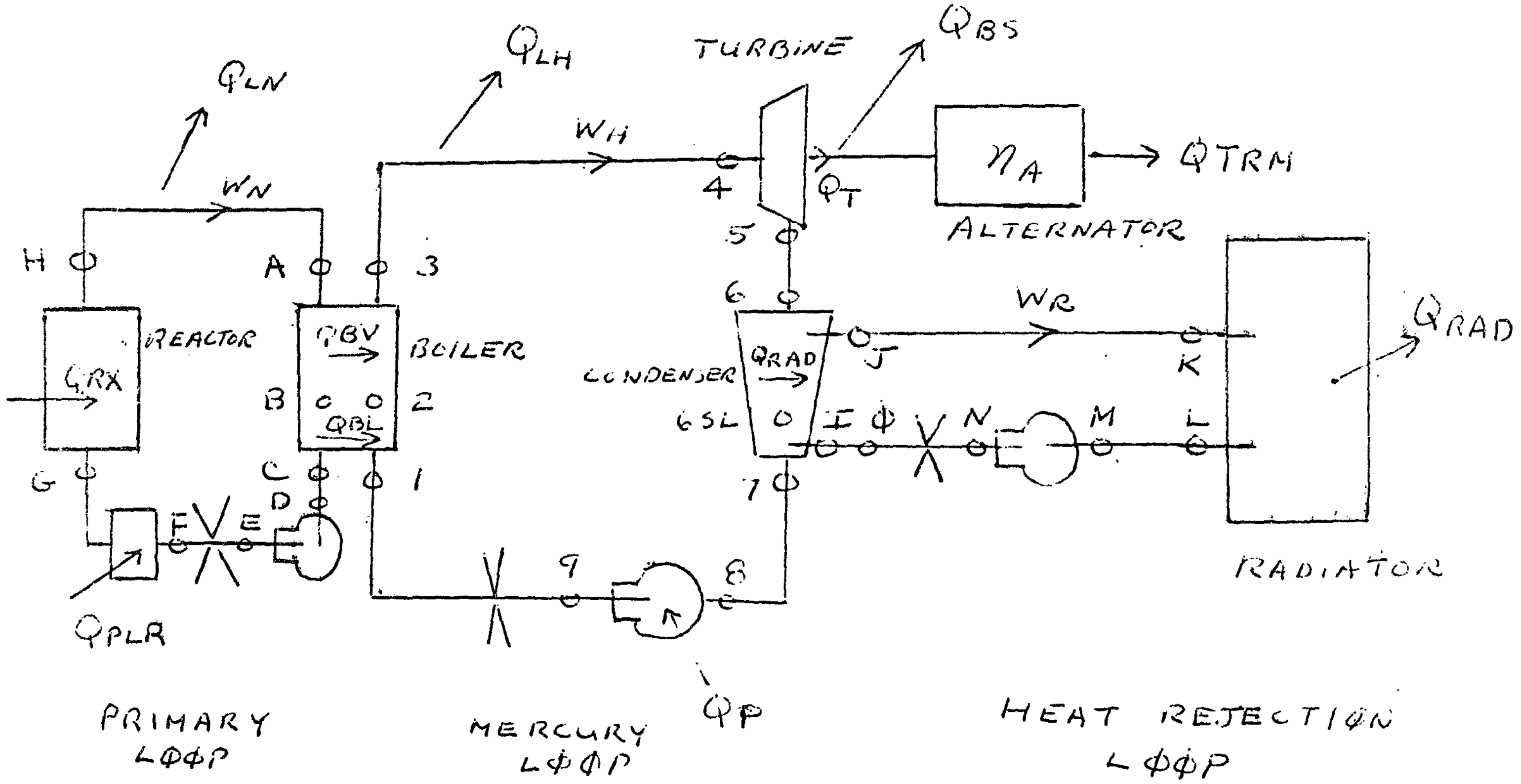


FIGURE 2
ALTERNATØR ØUTPUT LØADS

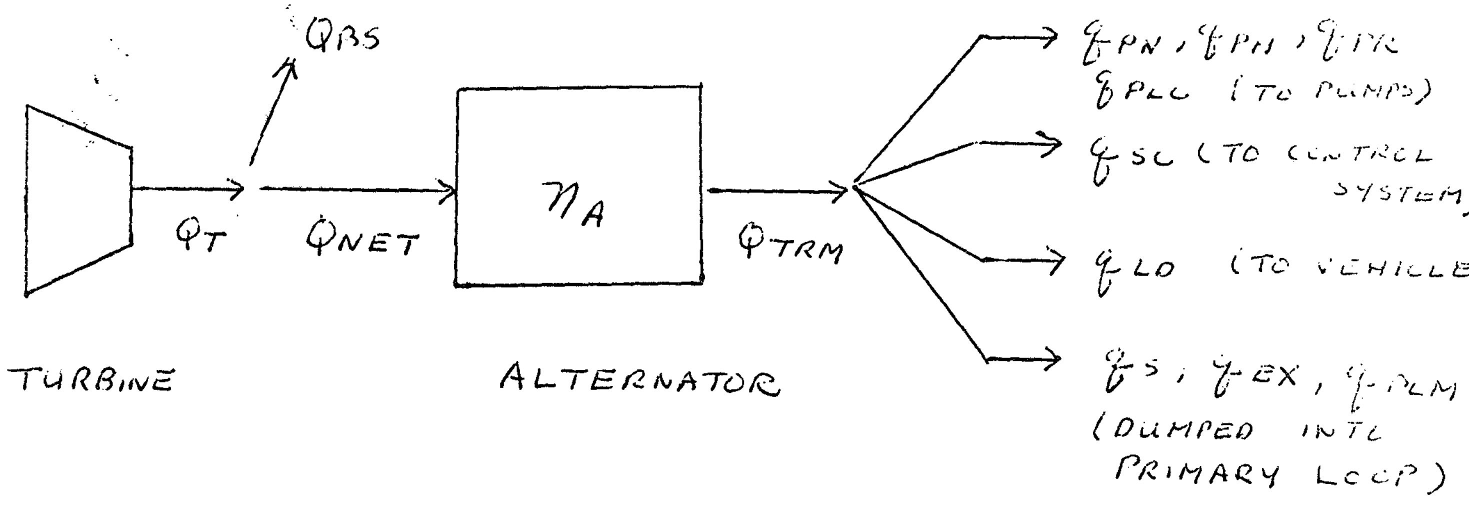


FIGURE 4
SAMPLE INPUT SHEET

```
      SCAN SAMPLE PRØB.WITH INFLUENCE COEFF. OPTION(VANDERBERG PERF.DATA TITLE
367.     139.     .224     .00525   3.25     .0117    7.    E-6.02534    N1-N8
.1769    .01428   .798     0.       8.2      4.35     1.5      .3175     N9-N16
407.     143.     .15      .0048    3.2      .0165    2.5   E-5.02534    C1-C8
.1769    .01428   7.25     6.4      5.4      4.55     4.93     .3535     C9-C16
6480.    94.86    3.3433   .3235    3.42     .025     .025     .2244     H1-H8
.2557    .02843   9.       56.      2070.    .4       0.       5.61      H9-H16
17.      .37      554.     1.144    .01964   4.616 E-5.523     .006      H17-LC6
0.       34.552   .02538   5.78  E-434.552   .02538   5.78  E-43.       LC7-PSFN
5.                                                                       PSFR
.17959   .025747  2.392 E-4.8768    .11188   .80844   .069191  9.0251E-4 G1-G8
48.052   2.252 E-31.7095E-66.375    .20625   .54496   5.8929E-35.9608E-5 G9-G16
.038302  2.1107E-32.468 E-411.649   .1196    .021062  1.14     .072      G17-G24
15.558   1.58     .058393  1806.1   146.58   .78882   9.1118E+5592.88    G25-G32
6.4487   138.18   .13056   1.2898E+414.059                               G33-G37
3.413    5.1549E-36.7545E-77.1329E-31.2429E-54.1184E-83.5508E-62.7188E-8 R1-R8
1.176E-10493.74   689.17                                                 R9-R11
.0252    .0324    .21      .212                                          CP1-CP4
 69 54 10 .001     1.0                                                  CØNTRØL
45900.   12000.   137.     41500.   4600.    .02      125.     1.74      X1-X8
110.     10.8     89.2     87.3     113.     4.29     2.86     4.39      X9-X16
1.36     .522     .725     .527     .613     7.8      140.     6.2       X17-X24
75.      1330.    2.       1.       3.       37.      35.      4.5       X25-X32
```

FIGURE 4 (CONT.)

```
54.      .75      .35      86.      .64      .86      66.      3.3       X33-X40
.54      162.     143.     1.27.    1.5      20.      .122     240.      X41-X48
16.      14.      6.       520.     50.      1330.    1170.    1140.     X49-X56
1150.    1340.    1.       1.3      450.     450.     490.     1100.     X57-X64
1310.    490.     320.     500.     670.                                 X65-X69
  3  4  5  6 25 28 29 31 34 40 45 49 51 54 59                            CΦL ID
```